

Diseño, desarrollo e implementación de una Plataforma Web para la gestión de servicios del Laboratorio

Miguel Ángel Ardila García, Juan Pablo Camacho Peñata, Daniel Santiago Parra Escobar, Alexandra Tinjacá Cortés

Facultad de Ingeniería
Universidad Piloto de Colombia
Bogotá D.C., Colombia

Emails: miguel-ardila@upc.edu.co, juan-camacho7@upc.edu.co, daniel-parra7@upc.edu.co, alexandra-tinjaca@upc.edu.co

Luis Felipe Herrera Quintero
Docente, Facultad de Ingeniería
Universidad Piloto de Colombia
Bogotá D.C., Colombia
Email: luis.herrera@upc.edu.co

Resumen—Este proyecto tiene como objetivo diseñar, desarrollar e implementar una plataforma web centralizada que optimice la gestión de reservas de salas de práctica libre y el préstamo de equipos en los laboratorios de la Universidad Piloto de Colombia. Se espera que la plataforma reduzca los tiempos de gestión, elimine el uso de formularios físicos, incremente la trazabilidad de los procesos y mejore la experiencia de usuarios (estudiantes, docentes y administradores).

Index Terms—gestión de laboratorios, plataforma web, reservas, préstamo de equipos, Universidad Piloto de Colombia.

I. INTRODUCCIÓN

En la actualidad, los procesos para separar salas de práctica libre y solicitar el préstamo de equipos en los laboratorios universitarios se realizan de manera manual, haciendo mucho uso de papel. Esto genera dificultades como pérdida de tiempo, problemas de disponibilidad y demás problemas técnicos.

La ausencia de un sistema centralizado ocasiona que estudiantes y docentes deban acudir presencialmente y muchas veces perder tiempo entre proceso y proceso en vez de solo llegar y recibir lo que se agendó previamente. Esto no sólo ralentiza la gestión, sino que también limita la eficiencia institucional y reduce la satisfacción de los usuarios [1].

II. ANTECEDENTES Y JUSTIFICACIÓN

II-A. Contexto actual del problema

En universidades de Colombia, se han adelantado procesos de modernización para digitalizar trámites académicos, principalmente a través de plataformas institucionales. Sin embargo, muchos de estos sistemas presentan limitaciones como:

- Interfaces poco amigables.
- Ausencia de reportes claros para la administración.
- Restricciones de acceso desde dispositivos móviles.

En el caso particular de nuestra institución, no existe aún una plataforma que permita a los estudiantes gestionar de forma virtual las reservas de salas y los préstamos de equipos de laboratorio, lo que genera demoras, baja eficiencia en la gestión académica y un alto consumo de recursos físicos [2].

II-B. Justificación

La implementación de un sistema web para la reserva de salas y préstamos de equipos se justifica en varios aspectos. En primer lugar, la digitalización de procesos manuales es un paso fundamental para las instituciones educativas que buscan optimizar recursos. Como señalan Hevner et al. (2004), “los sistemas de información que están bien diseñados y alineados con las necesidades del negocio pueden mejorar significativamente la eficiencia y la efectividad de los procesos organizacionales” [3].

En coherencia con la transformación digital de la educación superior, “un sistema accesible vía web permite a los estudiantes y docentes realizar sus solicitudes desde cualquier lugar, en cualquier momento” [4], rompiendo así las barreras de tiempo y ubicación. Finalmente, el sistema no solo automatiza un proceso, sino que genera un activo estratégico: datos. “El sistema almacenará de manera segura la información sobre reservas y préstamos, generando reportes claros que permitan tomar decisiones basadas en datos” [5], evitando conflictos en la asignación de recursos y facilitando la planificación institucional.

Aún si se utilizan formularios virtuales para la petición de salas de laboratorio para prácticas, la mayoría de los estudiantes requieren múltiples elementos y dispositivos que se deben pedir de manera presencial y llenando formularios manuales, lo que quita mucho tiempo, genera un volumen considerable de filas y tumultos y dificulta labores extras que

deben hacer los encargados de las zonas de laboratorios (como mantenimiento y arreglo correctivo a los dispositivos).

III. PLANTEAMIENTO DEL PROBLEMA

La gestión manual de reservas y préstamos en laboratorios genera pérdida de tiempo, problemas de disponibilidad, falta de trazabilidad y alto uso de papel. No existe un sistema centralizado que permita a los usuarios realizar estas gestiones de manera remota, lo que afecta la eficiencia operativa y la satisfacción de la comunidad universitaria.

IV. OBJETIVOS

IV-A. Objetivo principal

Generar una plataforma web que permita gestionar la separación de salas de práctica libre y la solicitud de préstamo de equipos en laboratorios, garantizando eficiencia, accesibilidad y trazabilidad de los procesos.

IV-B. Objetivos específicos

1. Analizar el estado del arte de sistemas de gestión de laboratorios y recursos académicos, identificando soluciones existentes (tanto comerciales como de código abierto), sus características técnicas, costos asociados y brechas que justifiquen el desarrollo de una solución personalizada para la universidad.
2. Diseñar la estructura técnica completa del sistema. Esto incluye crear el diagrama de la base de datos, hacer los bocetos de cómo se verán las pantallas principales y definir de manera clara y detallada cómo se comunicarán el frontend y el backend.
3. Implementar un producto mínimo viable (MVP) funcional que incluya los módulos de autenticación de usuarios por roles, reserva de salas, solicitud de préstamo de equipos y un panel administrativo básico, investigando y utilizando las tecnologías pertinentes.
4. Validar el correcto funcionamiento del sistema mediante la ejecución de pruebas unitarias, de integración y de usabilidad, asegurando que la plataforma cumpla con los requisitos funcionales y no funcionales establecidos, antes de su despliegue en un entorno de producción.

V. HIPÓTESIS

La implementación de una plataforma web centralizada reducirá significativamente el tiempo requerido para reservar salas y solicitar equipos, eliminará el uso de formatos físicos, mejorará la trazabilidad de los procesos y aumentará la satisfacción de los usuarios.

VI. ALCANCE

El presente proyecto tiene como delimitación fundamental la creación de una solución software de tipo plataforma web, cuyo propósito central es digitalizar y optimizar dos procesos básicos de la operación de los laboratorios: la reserva de espacios para prácticas libres y el préstamo de equipos e instrumentos. La plataforma actuará como un canal único y oficial para estas gestiones, reemplazando los métodos manuales basados en papel y planillas físicas.

VI-A. Inclusiones

El presente proyecto tiene como delimitación fundamental la creación de una solución software de tipo plataforma web, cuyo propósito central es digitalizar y optimizar dos procesos básicos de la operación de los laboratorios:

- Módulo de Gestión de Usuarios y Roles: El sistema distinguirá entre al menos tres perfiles de usuario con permisos diferenciados: estudiantes (quienes solicitan recursos), personal administrativo o de apoyo (quienes gestionan y aprueban las solicitudes) y administradores del sistema (con capacidades de configuración y supervisión global).
- Módulo de Gestión de Inventario: Se desarrollará un catálogo digital de todos los recursos disponibles para reserva y préstamo. Esto incluirá la capacidad de registrar, editar y deshabilitar tanto salas/laboratorios como equipos, describiendo sus características, estado y condiciones de uso.
- Módulo de Reservas y Solicitudes: Se implementará un sistema de autoservicio donde los usuarios podrán consultar en tiempo real la disponibilidad de los recursos, seleccionar franjas horarias específicas y formalizar sus solicitudes de reserva de espacios o préstamo de equipos a través de un flujo digital guiado.
- Módulo de Trazabilidad y Consulta: Toda solicitud generará un registro histórico auditable. Tanto usuarios como administradores podrán consultar el estado de sus trámites (pendiente, aprobado, rechazado, completado) y acceder a un historial de sus movimientos anteriores.
- Panel de Control Administrativo: Se proveerá una interfaz dedicada para que los gestores de laboratorios puedan aprobar o rechazar solicitudes, visualizar agendas consolidadas de uso de recursos, generar reportes básicos de utilización y gestionar incidencias.
- Despliegue y Accesibilidad web: La solución final será desplegada en un entorno de acceso web, garantizando que pueda ser utilizada desde cualquier dispositivo con un navegador moderno y conexión a internet, sin requerir instalación de software adicional por parte del usuario.

VI-B. Exclusiones

Quedan expresamente fuera del alcance del proyecto:

- Desarrollo de Aplicación Móvil Nativa: Si bien la plataforma será responsive y accesible desde móviles, el proyecto no incluye el desarrollo de aplicaciones nativas para iOS o Android.
- Integraciones Sistémicas Avanzadas: No se contempla, en esta fase inicial, la conexión con otros sistemas universitarios existentes (ej: plataforma de autenticación centralizada SIU, sistemas financieros para cobro de multas, software de inventario general de la universidad).
- Funcionalidades de Comunicación Interna: Características como notificaciones push automatizadas por correo electrónico o mensajería interna quedan fuera del alcance central.

- Mecanismos de Pago Digital: La gestión de cualquier tipo de cobro, fianza o multa monetaria asociada a los préstamos o reservas es considerada una funcionalidad de una posible fase posterior.
- Mantenimiento y Soporte Post-Implementación a Largo Plazo: El proyecto cubre el ciclo de vida de desarrollo hasta la puesta en producción y entrega oficial. El soporte técnico y mantenimiento correctivo deberá ser definido en un acuerdo aparte.
- Medidas de seguridad de los equipos físicos: EL proyecto no contempla la prevención de robos mediante cualquier acción física ni implementa medidas digitales relacionadas, pues esto depende de otros sectores universitarios que no tienen relación con el propósito original del proyecto de organizar, almacenar y usar información sobre los dispositivos y salas de laboratorio para el uso de la población estudiantil.

Esta delimitación clara del alcance permite enfocar todos los esfuerzos del equipo en el desarrollo de un producto mínimo viable (MVP) robusto y funcional, que resuelva el problema central, sentando las bases para posibles expansiones futuras una vez sea validado por la comunidad universitaria.

VII. METODOLOGÍA

Se seguirá una metodología ágil con enfoque incremental, basada en las fases de ingeniería de software. Esta elección se fundamenta en la naturaleza del proyecto, donde si bien los requisitos iniciales están definidos, existe una alta probabilidad de que estos evolucionen y se descubran necesidades adicionales durante el desarrollo. El marco ágil permite, mediante iteraciones cortas, “entregar software funcional con frecuencia, desde un par de semanas hasta un par de meses, con preferencia al período de tiempo más corto posible” [6].

Este enfoque es particularmente adecuado para “proyectos en los que los requisitos no se comprenden completamente en detalle de antemano y es probable que evolucionen” [7]. El proceso se estructurará en las siguientes fases, que se ejecutarán de manera incremental:

1. Análisis de requisitos (levantamiento de necesidades, casos de uso)
2. Diseño (arquitectura, modelo ER)
3. Implementación (desarrollo por módulos, integración continua)
4. Pruebas (unitarias, integración, usabilidad)
5. Despliegue (configuración de servidor, puesta en producción)

VII-A. Gestión de Calidad

La gestión de calidad garantiza que el producto cumpla con los requisitos establecidos y los estándares del proyecto. Incluye:

- Plan de Calidad: Documento que define los estándares, métricas y procedimientos de calidad.
- Revisiones Técnicas: Evaluaciones periódicas del código, diseño y documentación.

- Pruebas: Unitarias, de integración, sistema y aceptación para validar funcionalidades.

VIII. CRONOGRAMA

VIII-A. Fases del proyecto

VIII-A1. Fase 1: Definición de herramientas y modelos:

- BackEnd: Configurar repositorio y definir framework
- Bases de Datos: Levantar requerimientos y proponer MER
- FrontEnd: Crear wireframes iniciales y definir endpoints
- Documentación: Acta de inicio, plan de proyecto y casos de uso UML

VIII-A2. Fase 2: Definición de BackEnd:

- BackEnd: Crear proyecto base e implementar módulos de usuarios y reservas
- Bases de Datos: Implementar MER y conectar con framework via JPA
- FrontEnd: Validar endpoints y preparar diseños preliminares
- Documentación: Documentación técnica backend y coordinación de pruebas

VIII-A3. Fase 3: Implementación con FrontEnd:

- BackEnd: Extender lógica de negocio y ajustar endpoints
- Bases de Datos: Ajustar MER y apoyar pruebas de integración
- FrontEnd: Desarrollar vistas y conectar formularios con endpoints
- Documentación: Manual de usuario, diagramas UML y reportes de integración

VIII-A4. Fase 4: Integración y despliegue:

- BackEnd: Pruebas unitarias y ajustes de controladores
- Bases de Datos: Validar integridad y optimizar consultas
- FrontEnd: Pruebas manuales de flujo completo
- Documentación: Documentación técnica final, manual de usuario y despliegue

VIII-B. Cronograma de Actividades por Semanas

Semana 1 (2-5 de septiembre de 2025):

- BackEnd: Configurar repositorio y definir configuración del framework
- Bases de Datos: Levantar requerimientos de datos iniciales
- Documentación: Redactar Plan del Proyecto

Semana 2 (8-12 de septiembre de 2025):

- Procesos Técnicos: Definir requisitos detallados del sistema
- Bases de Datos: Proponer borrador del modelo entidad-relación (MER)
- FrontEnd: Crear wireframes iniciales (login, reservas, listados)
- Documentación: Crear Plan de Proyecto preliminar con cronograma
- Procesos de Soporte: Definir plan de gestión de calidad

Semana 3 (15-19 de septiembre de 2025):

- Bases de Datos: Instalar y configurar base de datos en servidor local
- FrontEnd: Definir endpoints para validación con backend
- Documentación: Redactar casos de uso UML de actores

Semana 4 (22-26 de septiembre de 2025):

- Procesos Técnicos: Iniciar diseño arquitectónico detallado
- BackEnd: Crear proyecto base con controllers, services, repositories
- BackEnd: Implementar módulo de usuarios (login, roles, autenticación)

Semana 5 (29 septiembre - 3 octubre de 2025):

- Procesos Técnicos: Continuar implementación de módulos centrales
- BackEnd: Implementar módulo de reservas de laboratorio (crear, listar)
- Bases de Datos: Implementar modelo entidad-relación en base de datos
- Bases de Datos: Conectar base de datos con framework mediante JPA
- Procesos de Verificación: Realizar pruebas iniciales de integración

Semana 6 (6-10 de octubre de 2025):

- FrontEnd: Validar endpoints creados con backend
- FrontEnd: Preparar diseño preliminar para login y reservas
- Documentación: Redactar documentación técnica preliminar del backend
- Procesos de Soporte: Actualizar plan de gestión de riesgos

Semana 7 (13-17 de octubre de 2025):

- Procesos Técnicos: Extender funcionalidades del sistema
- BackEnd: Extender lógica de negocio para préstamos y devoluciones
- BackEnd: Ajustar endpoints para integración con frontend
- Bases de Datos: Ajustar modelo entidad-relación según feedback
- Procesos de Verificación: Ejecutar pruebas de integración BD-backend

Semana 8 (20-24 de octubre de 2025):

- Procesos Técnicos: Integración frontend-backend
- FrontEnd: Desarrollar vistas completas del sistema
- FrontEnd: Implementar formularios (registro, login, solicitudes)
- FrontEnd: Conectar formularios con endpoints del Backend
- Documentación: Redactar manual preliminar de usuario

Semana 9 (27-31 de octubre de 2025):

- Procesos de Verificación: Ejecutar pruebas integrales del sistema
- Documentación: Documentar flujos del sistema con diagramas UML

- Documentación: Preparar reporte de integración semanal
- Procesos de Soporte: Realizar auditoría de calidad

Semana 10 (3-7 de noviembre de 2025):

- Procesos de Verificación: Ejecutar pruebas unitarias exhaustivas
- BackEnd: Ajustar controladores y servicios según resultados de pruebas
- Bases de Datos: Validar integridad y consistencia de datos
- Bases de Datos: Optimizar consultas (índices, relaciones)
- Procesos de Transición: Preparar entorno de producción

Semana 11 (10-14 de noviembre de 2025):

- Procesos de Transición: Desplegar sistema en entorno de producción
- FrontEnd: Realizar pruebas manuales de flujo completo
- Documentación: Compilar documentación técnica final
- Documentación: Redactar manual de usuario final y entrega paper proyecto final.
- Procesos de Acuerdo: Entrega formal y aceptación del sistema
- Procesos de Evaluación: Realizar revisión final del proyecto

IX. ROLES Y RESPONSABILIDADES

Se presentan los siguientes roles y su correspondiente distribución y obligaciones en el proyecto.

Cuadro I: Distribución de roles del equipo

Rol	Responsable
Desarrollador Frontend	Miguel Ardila
Desarrollador BackEnd	Juan Pablo Camacho Peñata
Administrador de Bases de Datos	Daniel Santiago Parra Escobar
Líder de Proyecto / Ingeniero de requerimientos	Alexandra Tinjacá Cortés

IX-A. Responsabilidades principales por área

Área: Requerimientos

- Actividades principales: Levantamiento de necesidades, documentación y validación de requisitos.
- Responsable(s): Alexandra (Documentación), con apoyo del equipo

Área: Diseño

- Actividades principales: Definición de arquitectura, diseño de interfaz y modelado de BD.
- Responsable(s): Miguel (Frontend), Santiago (BD), supervisión Alexandra

Área: Técnicos

- Actividades principales: Desarrollo Spring Boot, interfaz web, integración y pruebas unitarias.
- Responsable(s): Juan Pablo (Backend), Miguel (Frontend), Santiago (BD)

Área: Verificación

- Actividades principales: Pruebas funcionales, de rendimiento y documentación de incidencias.
- Responsable(s): Alexandra (Líder), con apoyo del equipo

Área: Gestión de Configuración

- Actividades principales: Git/GitHub, documentación de cambios y manejo de ramas.
- Responsable(s): Juan Pablo (Backend), con apoyo del equipo

Área: Gestión de Proyecto

- Actividades principales: Cronograma, seguimiento de tareas y coordinación.
- Responsable(s): Alexandra (Líder)

Área: Gestión de Calidad

- Actividades principales: Estándares de código, revisiones y cumplimiento de requisitos.
- Responsable(s): Todo el equipo, supervisado por Alexandra

X. PRESUPUESTO DETALLADO

Este apartado presenta el desglose detallado de costos para el desarrollo e implementación del sistema de gestión de laboratorios durante un período de tres meses. Los costos se han mensualizado cuando es posible, excepto para ítems que requieren pago anual obligatorio. El presupuesto considera costos de infraestructura en la nube, salarios del equipo de desarrollo junior y costos adicionales de contingencia. Cabe recalcar que estos valores y todo entorno a la toma de decisión y manejo de los servicios presupuestados se realiza en el Anexo B. Téngase en cuenta que se hizo un análisis exhaustivo del framework a usar en este proyecto como se evidencia en el Anexo A; la selección del framework se basó en criterios de escalabilidad y mantenibilidad.

X-A. Costos de Infraestructura

Cuadro II: Costos Mensuales de Infraestructura

Item	Frecuencia	Costo Mensual (COP)	Total 3 meses (COP)
Supabase Starter	Mensual	\$100,000	\$300,000
Render (2 servicios)	Mensual	\$56,000	\$168,000
Netlify Pro	Mensual	\$76,000	\$228,000
Dominio (.tech)	Anual	\$10,700	\$32,000
Subtotal		\$242,700	\$728,000

X-B. Costos de Recursos Humanos

Cuadro III: Costos de Recursos Humanos (3 meses)

Rol	Salario Mensual (COP)	Meses	Total (COP)
Desarrollador Full-Stack	\$2,500,000	3	\$7,500,000
Desarrollador Frontend	\$2,500,000	3	\$7,500,000
Desarrollador Backend	\$2,500,000	3	\$7,500,000
Desarrollador BD	\$2,500,000	3	\$7,500,000
Subtotal			\$30,000,000

X-C. Resumen General de Costos

Cuadro IV: Resumen de Costos Totales (3 meses)

Concepto	Total COP (3 meses)
Infraestructura	\$728,000
Recursos Humanos	\$30,000,000
Subtotal	\$30,728,000
Fondo de contingencias (5 %)	\$1,536,400
Total General	\$32,264,400

X-D. Notas Presupuestales

- Los costos de infraestructura se basan en planes básicos de cada servicio durante el período de desarrollo.
- El dominio requiere pago anual por adelantado, por lo que se incluye el prorrato mensual.
- Los salarios corresponden a desarrolladores junior con experiencia de 0-2 años en el mercado colombiano.
- El fondo de contingencias del 5 % cubre imprevistos técnicos o sobrecostos inesperados.
- Los costos de mantenimiento post-desarrollo serán inferiores al no requerir el equipo completo de desarrollo.

X-E. Potencial de escalabilidad y sostenibilidad

El sistema está diseñado con una arquitectura modular que facilita la adición de nuevas funcionalidades en el futuro, como reportes avanzados y analítica de uso, y extensión a otros servicios académicos de la universidad. El uso de tecnologías abiertas y ampliamente documentadas garantiza mantenimiento y actualización sostenible.

XI. CONCLUSIONES

XI-A. Impacto esperado en la universidad

El desarrollo e implementación de la plataforma web permitirá optimizar la gestión de recursos en los laboratorios, agilizando los procesos de reserva de salas y préstamo de equipos. Esto reducirá tiempos de espera, aumentará la eficiencia operativa y mejorará la experiencia de estudiantes y docentes.

ANEXO A: PREGUNTAS FRECUENTES (FAQ)

Este anexo recopila preguntas frecuentes y cuestionarios de validación realizados durante la fase de análisis y diseño del proyecto. Cada pregunta se responde haciendo referencia a la sección del documento donde se aborda la solución o el enfoque propuesto.

XI-B. Seguridad y Protección de Datos

- ¿Qué mecanismos implementarán para garantizar la seguridad y protección de la información de los estudiantes?
- ¿Qué medidas implementarán para garantizar la protección y confidencialidad de la información de los estudiantes?
- ¿Cómo manejarán la seguridad de información de los estudiantes?
- ¿Cómo van a cuidar la información de los estudiantes que se guarde en la plataforma?

- ¿Cómo manejaran los permisos y la seguridad de los implementos prestados con la plataforma?
- ¿Quién maneja el tratamiento de datos?

Respuesta: La seguridad de los datos se garantiza mediante múltiples capas. En la capa de base de datos, se utiliza **Row Level Security (RLS)** de Supabase, que restringe el acceso a los datos basándose en roles de usuario, como se detalla en el *Documento Técnico de Presupuesto y Análisis Técnico* (Sección IV-A y IV-D). Para la autenticación, se emplean **tokens JWT** (JSON Web Tokens) generados por Supabase y validados por el backend en Spring, que también incluye Spring Security para gestionar autorizaciones y permisos (véase el *Informe Técnico de Selección de Framework*, Sección V-B). Todas las comunicaciones se realizan sobre HTTPS para encriptar los datos en tránsito. El diseño modular de roles (estudiante, administrador, personal de apoyo) asegura que cada usuario solo acceda a la información necesaria para sus funciones, como se describe en el *Plan de Proyecto* (Sección VI-A).

XI-C. Integración con Sistemas Universitarios

- ¿Cómo integrarán a los usuarios en la gestión académica y administrativa de los estudiantes dentro de la universidad?
- ¿Cómo planean vincular de forma efectiva a los usuarios dentro del sistema de gestión de estudiantes de la universidad?
- ¿Cómo integraran los usuarios a la gestión de estudiantes de la universidad?

Respuesta: La integración se logra mediante el **Módulo de Gestión de Usuarios y Roles**, definido en el *Plan de Proyecto* (Sección VI-A). Este módulo permite autenticar usuarios con roles específicos (estudiante, administrador, personal de apoyo) y asignar permisos diferenciados. Inicialmente, la plataforma operará de forma independiente, pero su arquitectura está diseñada para permitir futuras integraciones con sistemas universitarios existentes (como SIU) mediante APIs RESTful. El uso de Spring Boot facilita la creación de endpoints seguros y escalables para dichas integraciones, como se justifica en el *Informe Técnico de Selección de Framework* (Sección VI).

XI-D. Escalabilidad y Uso en Otras Universidades

- ¿Esta plataforma solo se podrá usar para la universidad piloto o se podría utilizar para otras universidades?
- ¿El proyecto está solo destinado para la universidad piloto?
- ¿Cómo manejarán la escalabilidad si en un futuro la plataforma se quiere implementar en varias universidades?
- ¿Se puede implementar a algún otro servicio fuera de los laboratorios en un futuro?
- ¿Cuál será la alcanzabilidad a futuro del proyecto?

Respuesta: Aunque el desarrollo inicial está orientado a la Universidad Piloto de Colombia, la arquitectura del sistema es **modular y escalable**, lo que permitiría su adaptación a otras instituciones. Esto se menciona en el *Plan de Proyecto* (Sección X-E). La elección de tecnologías como Spring (para backend, más específicamente SpringBoot) y Supabase

(para base de datos) asegura que el sistema pueda manejar un crecimiento en usuarios y funcionalidades sin cambios drásticos. El diseño basado en microservicios y APIs RESTful facilita la extensión a otros servicios académicos en el futuro. El *Documento Técnico de Presupuesto* (Sección IV) detalla la capacidad de escalabilidad de las plataformas en la nube seleccionadas (Render, Supabase Pro). Además en el apartado de alcance (Sección VI) en el *Plan de Proyecto* se muestra que el proyecto está destinado solamente a la universidad Piloto de Colombia en la duración de este proyecto, pero dejando abierta la posibilidad de migrar también a otras universidades por su construcción en módulos, este proyecto además y como se menciona en la sección (Sección VI-A) contempla únicamente la sección de laboratorios, pero por las mismas razones ya mencionadas se podría llegar a ampliar a la zona de biblioteca estudiantil.

XI-E. Costos y Financiamiento

- ¿Como usuario se debe realizar algún pago, si es así sería semestral o mensual?
- ¿De donde van a sacar el dinero?
- ¿Van a usar solo herramientas gratis o después necesitarán pagar por algunas?
- ¿Las herramientas gratis son capaces de cubrir la demanda inicial del proyecto o serán solo para pruebas y después pasar a la de pagos?

Respuesta: El proyecto no contempla ningún **mecanismo de pago por parte de los usuarios** en esta fase, como se establece en las exclusiones del *Plan de Proyecto* (Sección VI-B). Los costos de desarrollo e infraestructura están cubiertos por el presupuesto asignado, que incluye planes de pago para servicios en la nube (Supabase Pro, Render Paid, Netlify Pro) durante la fase de despliegue, como se detalla en el *Documento Técnico de Presupuesto* (Sección V-A). La inversión inicial del equipo se detalla en el presupuesto a la hora de mencionar los servicios elegidos (como supabase y render), que son de pago y serán vitales a la hora de realizar pruebas y hacer el despliegue. Las herramientas gratuitas se utilizan para desarrollo inicial, pero se escalan a planes pagos para garantizar rendimiento en producción. El fondo de contingencias del 5% cubre imprevistos (Sección V-D del mismo documento). Además también menciona lo que cubren las plataformas en cuestión de TPS, y cubren más de las que se tienen presupuestadas al usar las opciones de pago.

XI-F. Control de Inventario y Estado de Equipos

- ¿A la hora de entregar una sala si se daña un dispositivo quien es el encargado de actualizar o eliminar y como se sabe si se daño por tiempo o por el mal uso del estudiante?
- ¿La universidad cuenta con diversas salas como van a ingresar cada objeto en la sala y el estado de este?
- ¿Quién va a gestionar el inventario de los laboratorios?
- ¿Como gestionarán el manejo y evitar posibles robos de los materiales de trabajo?

Respuesta: El **Módulo de Gestión de Inventario** (descrito en el *Plan de Proyecto*, Sección VI-A) permite registrar, editar y deshabilitar salas y equipos, incluyendo su estado y condiciones de uso. Los administradores de laboratorio son responsables de actualizar el inventario a través del **Panel de Control Administrativo**, que incluye funcionalidades para reportar daños o incidentes. El sistema registra el historial de uso de cada equipo, lo que ayuda a determinar si un daño fue por mal uso o por tiempo. Sin embargo, la verificación física del estado de los equipos sigue siendo necesaria y queda fuera del alcance del software. La prevención de robos se maneja mediante registros de préstamos y devoluciones, pero no incluye medidas físicas de seguridad, así como se especifica claramente en el *Plan de Proyecto* (Sección VI-B) que habla de las exclusiones del proyecto.

XI-G. Funcionalidades de la Plataforma

- ¿Esta plataforma solo se podrá usar para el pedido de salas o también para los dispositivos que nos ofrece la universidad?
- ¿Cada que vaya a pedir algún instrumento tengo que hacer el mismo proceso? ¿Hay algún límite o control para esto?
- ¿Qué reglas aplican (máxima duración, antelación mínima, prioridad a docentes sobre estudiantes)?
- ¿Cómo evitan doble reserva?
- ¿Es un 100% lo que tienen pensado eliminar el papel en el proyecto?
- ¿Si un estudiante presenta dos peticiones tiene que hacerlo dos veces o simplemente actualiza la información?
- ¿Qué asegurará el cuidado de los implementos prestados?

Respuesta: La plataforma gestiona tanto **reservas de salas** como **préstamos de equipos**, según se especifica en el *Plan de Proyecto* (Sección IV-A). El proceso de solicitud es digital y estandarizado, pero los usuarios deben realizar una solicitud por cada recurso o franja horaria. El sistema incluye reglas configurables (límites de tiempo, prioridades) que son aplicadas automáticamente por el backend y que se ajustarán también a las necesidades de los administradores y la universidad. La **doble reserva se evita** mediante validaciones en tiempo real de la disponibilidad en la base de datos. El objetivo es eliminar el 100% del uso de papel para estos procesos, aunque puede haber casos excepcionales no cubiertos. El cuidado de los implementos se asegura mediante registros de responsabilidad y historial de uso, pero la supervisión física sigue siendo necesaria.

XI-H. Adopción y Capacitación

- ¿Qué nivel de aceptación tendría una herramienta digital entre estudiantes, docentes y personal administrativo de la universidad?
- ¿Qué barreras (tecnológicas, culturales o administrativas) pueden dificultar la adopción de la plataforma en la comunidad universitaria?
- ¿Qué estrategias usarán para capacitar a docentes y administrativos en el uso de la herramienta?

- ¿Qué estrategias usarán para asegurar que los estudiantes realmente usen la plataforma y no recurran al sistema actual manual?
- ¿Cómo van a convencer a los laboratorios para la cambiar un simple código QR a tener que gastar en una página?
- ¿Cómo retroalimentarán los usuarios que usen la plataforma?

Respuesta: La adopción se fomentará mediante **capacitaciones y material de apoyo** (como el manual de usuario mencionado en el *Plan de Proyecto*, Sección VIII-A4). Se espera una alta aceptación debido a la conveniencia de la gestión digital. Las barreras potenciales (como resistencia al cambio o falta de conectividad) se mitigarán con una interfaz intuitiva y responsive. La retroalimentación se recopilará a través de canales digitales dentro de la plataforma. La justificación económica se basa en el ahorro a largo plazo al reducir el papel y mejorar la eficiencia, como se argumenta en la *Justificación* (Sección II-B del Plan de Proyecto). Si bien se utiliza un qr, este es únicamente para la petición de salas para prácticas y no asegura que se pueda ingresar rápidamente ni evita llenar formularios físicos en sala para petición de otros elementos que podrían requerirse dentro de la práctica, haciendo perder tiempo valioso que podría ahorrarse solicitando todo con anterioridad, todo esto argumentado en la misma sección que se acaba de referenciar.

XI-I. Aspectos Técnicos y Rendimiento

- ¿Qué protocolos usarán para evitar las caídas?
- ¿Qué proyectos usaron para el estudio del arte?
- ¿En qué parte se utilizarán las bases de datos?
- ¿Qué lenguaje de programación utilizarán?
- ¿Cómo van a garantizar que la plataforma web se mantenga segura al manejar datos de inventario, reservas y pagos?
- ¿Cómo planear manejar si el número de usuarios es superado al que dijeron?
- ¿Si se cae la página o no hay Internet como llevarán los controles?
- ¿Si se cae la página y necesitan implementos, qué harían?
- ¿Cómo identificará físicamente cada equipo (QR, código de barras, tag RFID o solo texto)?

Respuesta: Los **protocolos para evitar caídas** incluyen el uso de servicios en la nube con alta disponibilidad (Render, Supabase) y técnicas de optimización como caching y compresión, como se detalla en el *Documento Técnico de Presupuesto* (Sección III). El estudio del estado del arte se basó en análisis de frameworks y plataformas, documentado en el *Informe Técnico de Selección de Framework* (Sección IV). La base de datos se utiliza para almacenar todos los datos persistentes (usuarios, recursos, reservas) y se implementa con Supabase (PostgreSQL). El lenguaje de programación principal es Java con Spring Boot para el backend, y JavaScript/HTML/CSS para el frontend. La escalabilidad está garantizada por la arquitectura en la nube (Sección IV del Documento Técnico de Presupuesto). En caso de caída, se tiene un plan de contingencia que incluye revertir a procesos

manuales temporales, sin embargo se tiene a consideración la creación de guías e informes para la contingencia y control ante posibles problemas como también lo solicita la ISO 15288. La identificación física de equipos queda fuera del alcance del software, pero se podría integrar en fases futuras, por el momento se contempla obtener un inventario por parte de la universidad en las fases más profundas del proyecto.

XI-J. Preguntas Varias

- ¿Los módulos que usarán los desarrollaran ustedes mismos?
- ¿Por qué la universidad debería elegirlos a ustedes y no el sistema actual, que aunque es mas sencillo es práctico y funciona?
- ¿Cómo van a hacer para que la universidad acepte esto?
- ¿Qué problemas se generan actualmente por la gestión manual de salas y equipos en los laboratorios universitarios?
- ¿Cuáles son los objetivos específicos del desarrollo de la plataforma web para la gestión de servicios del laboratorio?
- ¿Cómo van a llevar el control?
- ¿Cuáles son las ventajas y desventajas del uso de la plataforma web en la UPC?
- ¿Ustedes seguirán manteniendo la página web o lo remiten a los laboratoristas?

Respuesta: Todos los módulos son **desarrollados internamente** por el equipo del proyecto, como se describe en los roles y responsabilidades (*Plan de Proyecto*, Sección IX). La ventaja sobre el sistema manual se justifica por la reducción de tiempo, eliminación de papel y mejora en la trazabilidad (Sección I y II-B del Plan de Proyecto), además como se menciona en la sección de justificación (Sección II-B del Plan de Proyecto), el valor más grande de este proyecto no es solo la organización, son los datos y el valor monetario que podría reflejarse en la universidad, a la hora de invertir en equipos, mantenimiento, salas, etc, tener un control más eficaz y con trazabilidad genera una cantidad de datos inmensa y que muestra qué se usa y cuánto se usa, cosa que puede ser utilizada para mejorar la vida de los empleados y de los estudiantes. La aceptación por la universidad se basará en demostraciones del MVP y los beneficios tangibles. Los problemas actuales incluyen pérdida de tiempo, falta de trazabilidad y alto uso de papel (Sección III). Los objetivos específicos se listan en la Sección IV-B del Plan de Proyecto. El control se lleva mediante registros digitales y paneles administrativos. Las ventajas incluyen eficiencia y accesibilidad; las desventajas podrían ser la curva de aprendizaje inicial (a pesar de mencionarse en varias ocasiones en el cronograma, sección VIII, la creación de manuales de usuario). El mantenimiento post-desarrollo será transferido a la universidad, como se indica en las exclusiones (Sección VI-B).

ANEXOS

Anexo A: Informe Técnico - Selección del Framework Backend para el Sistema de Gestión de Laboratorios

Documento técnico interno del equipo de desarrollo que detalla el proceso de evaluación, criterios de selección y justificación técnica del framework backend elegido para el sistema de gestión de laboratorios.

Documento de referencia: INF-TEC-001-v1

Anexo B: Documento Técnico - Presupuesto y Análisis Técnico para un Sistema de Gestión de Laboratorios con Arquitectura Escalable en la Nube

Documento técnico que especifica la arquitectura del sistema, componentes, tecnologías empleadas y la estrategia de despliegue en entorno de producción.

Documento de referencia: INF-TEC-002-v1

REFERENCIAS

- [1] M. Fowler, *Patterns of Enterprise Application Architecture*. Boston, MA: Addison-Wesley, 2002.
- [2] L. Richardson, M. Amundsen, and S. Ruby, *RESTful Web APIs*. Sebastopol, CA: O'Reilly Media, 2013.
- [3] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, pp. 75-105, Mar. 2004.
- [4] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York, NY: McGraw-Hill Education, 2020.
- [5] K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, 15th ed. Upper Saddle River, NJ: Pearson, 2016.
- [6] K. Beck et al., "Manifesto for Agile Software Development," 2001. [Online]. Available: <http://agilemanifesto.org/>
- [7] I. Sommerville, *Software Engineering*, 9th ed. Boston, MA: Addison-Wesley, 2011.
- [8] B. W. Boehm, *Software Engineering Economics*. Englewood Cliffs, NJ: Prentice-Hall, 1981.
- [9] M. Hamdaqa and L. Tahvildari, "Cloud computing uncovered: A research landscape," in *Advances in Computers*, vol. 86, 2012, pp. 41-85.
- [10] Supabase, Inc., "Supabase Pricing," 2023. [Online]. Available: <https://supabase.com/pricing>
- [11] Google LLC, "Google Trends: Spring Boot, Express.js, NestJS, Jakarta EE," 2023. [Online]. Available: <https://trends.google.com/>
- [12] Stack Overflow, "Stack Overflow Developer Survey 2023," 2023. [Online]. Available: <https://survey.stackoverflow.co/2023/>